Denver Law Review

Volume 71
Issue 3 *Symposium - Software Issue in Computer Law*

Article 10

January 2021

Computer Copyright Protection Narrows as Video Game Giants Battle in Atari v. Nintendo

Susan E. Dallas

COMPUTER COPYRIGHT PROTECTION NARROWS AS VIDEO GAME GIANTS BATTLE IN *Atari v. Nintendo*

INTRODUCTION

One out of every three households in the United States has one, and the next version will cost only $49.95. What is it? The Nintendo home video game system.[1] Since its introduction in 1985, the Nintendo Entertainment System (NES) has become the number one video game system in the world.[2] Video games represent only one facet of the computer industry, which has fostered tremendous advances in technological and economic growth in the past twenty-five years.[3] As a result of this expansion, the United States has witnessed a growing number of intellectual property disputes.[4] Today, industry leaders operate in fear of litigation over infringement of patents and copyrights, and of unfair business practices which may deny risk-taking companies the benefits of their creativity.[5] Although copyright laws do protect computer programs,[6] a recent lawsuit involving two video game giants suggests that this protection has been narrowed.[7]

In *Atari Games Corp. v. Nintendo of America Inc.*,[8] the U.S. Court of Appeals for the Federal Circuit sustained a preliminary injunction, enjoining Atari Games Corporation (Atari) from infringing Nintendo's copyright of its 10NES computer program.[9] The court addressed the extent of protection given to a computer program and attempted to clarify to what extent a programmer may and may not use another's work to write a program.

---

1. *More Can Buy the Most Popular Video Game System in History; 8-bit NES Gets New Chassis and Price Tag Below $50*, BUSINESS WIRE, June 3, 1993, *available in* LEXIS, Nexis Library, BWIRE File.

2. Nintendo has sold a total of 60 million hardware units worldwide, and occupies a 99 percent share of the 8-bit market in the U.S. In 1992, the NES accounted for $2.7 million in hardware sales and almost $1 billion in software sales. *Id.*

3. In the mid 1960s, sales of computer programs generated approximately $250 million in annual sales. COMPUTER SCIENCE AND TELECOMMUNICATIONS BOARD, NATIONAL RESEARCH COUNCIL, INTELLECTUAL PROPERTY ISSUES IN SOFTWARE vii (1991) [hereinafter "CSTB"]. Since then, the number of U.S. software firms has quadrupled, generating tens of billions of dollars in revenues annually. In 1990, sales of packaged software alone by U.S. firms totaled nearly $20 billion and occupied more than 40% of the world market. *Id.* at 3.

4. Computers have even made theft of intelligence easier, since employees can copy computer files which may contain trade secrets. *See* Jolie Solomon *et al.*, *The Grand Pilferer?*, NEWSWEEK, June 7, 1993, at 38, 39.

5. CSTB, *supra* note 3, at ix.

6. Copyright Act of 1976, 17 U.S.C. §§ 101 to 1010 (1988 & Supp. IV 1992). *See also* 1 MELVILLE B. NIMMER & DAVID NIMMER, NIMMER ON COPYRIGHT § 2.04[C] (1993).

7. *See* David R. Ellis, *Chips, Locks, and Video Games*, FLA. BAR J., July-Aug. 1993, at 75, 79; Lee T. Gesmer, *Decisions May Signify A Judicial Turnabout*, NAT'L L. J., Jan. 18, 1993, at S2, S4; Philip J. McCabe & William A. Tanenbaum, *Copyright Decisions Increase the Value of Patent Protection for Computer Software*, 2 J. PROPRIETARY RTS. 2, *available in* WESTLAW, JLR database, (1993).

8. 975 F.2d 832 (Fed. Cir. 1992).

9. *Id.* at 847.

This Comment examines the court's decision in *Atari*. First, the Comment briefly introduces computer technology to facilitate the discussion. Second, the Comment explores the doctrines and cases relating to copyright law, particularly its development since the 1970s. Third, the Comment explains the facts and the reasoning adopted by the *Atari* court. Finally, the Comment analyzes *Atari*, noting its likely impact on the computer industry in the future.

I. BACKGROUND

A. *Introduction to Computer Technology*

Computer technology has progressed significantly since the first computer was built in 1944.[10] Today, the software[11] and computer programs which make the industry so lucrative[12] are, perhaps, more important than the computer itself. A computer program is defined by statute as "a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result."[13] Programmers normally write software programs in source code, a high-level computer language readable by humans.[14] Since computers are incapable of understanding commands expressed in source code, the computer contains an assembler or compiler program which translates the source code into object code[15], a binary code[16] which is readable by computers. The computer stores the object code on disks or silicon chips.[17]

The two principle types of computer programs are operating systems and application systems.[18] An operating system is a series of programs which manages the internal functions of the computer, while an application system is a program which performs tasks for the user.[19] Video game

---

10. Peter D. Aufrichtig, Note, *Copyright Protection for Computer Programs in Read Only Memory Chips*, 11 HOFSTRA L. REV. 329, 333 (1982).

11. Software includes "project description and research, source code, object code, program documentation, user instructions, and operating manuals." GORDON V. SMITH & RUSSELL L. PARR, VALUATION OF INTELLECTUAL PROPERTY AND INTANGIBLE ASSETS 115 (1989).

12. The computer software industry is a multi-billion dollar business with millions of users. ANTHONY L. CLAPES, SOFTWARE, COPYRIGHT, AND COMPETITION 19 (1989).

13. 17 U.S.C. § 101 (1988).

14. Standard programming languages which are examples of source code are FORTRAN and COBOL. Howard Root, Note, *Copyright Infringement of Computer Programs: A Modification of the Substantial Similarity Test*, 68 MINN. L. REV. 1264, 1266 (1984). These high-level languages are powerful because each statement usually performs the same task as several machine language (object code) instructions. OFFICE OF TECHNOLOGY ASSESSMENT, CONGRESS, FINDING A BALANCE: COMPUTER SOFTWARE, INTELLECTUAL PROPERTY AND THE CHALLENGE OF TECHNOLOGICAL CHANGE 130 (1992) [hereinafter "CONGRESS"].

15. Aufrichtig, *supra* note 10, at 340 n.110.

16. Binary form is a system of numeration consisting of ones and zeroes, each representing an open or closed circuit in the computer. Root, *supra* note 14, at 1267 n.16. Inside a computer, both data and instructions are represented by patterns of electronic signals which take one of two values, a "1" or a "0." For example, in most computers, the letter "A" is represented by a pattern of signals which corresponds to "01000001." CONGRESS, *supra* note 14, at 125.

17. Root, *supra* note 14, at 1267.

18. Peter S. Menell, *An Analysis of the Scope of Copyright Protection for Application Programs*, 41 STAN. L. REV. 1045, 1051 (1989).

19. *Id.*

programs are typically application programs.[20] In addition, the complexity of many home and arcade video games requires the use of microprocessors. A microprocessor is a miniature computer on a single chip[21] within the computer, which processes information by executing a series of instructions.[22]

Most programmers create software with a "top down design" process. This process involves defining the problem or function for the program, decomposing this function by defining the tasks and subtasks for the computer to perform, and writing instructions or commands for the computer in the form of source code.[23] The computer's assembler program then compiles the instructions, enabling the computer to perform the tasks.[24] Programmers can also reverse this process to obtain source code from a program on a disk or chip. This process, called "reverse engineering" or "intermediate copying"[25] involves using a "disassembly" program[26] to translate object code into source code.[27] The programmer then reviews the code to understand the program's input, output, and functions, and generates a set of specifications describing the program.[28] A second programmer then generates a new program using the first programmer's specifications.[29] This second programmer works in a "clean room,"[30] and never sees the original code used to create the specifications.[31] Since this process can eliminate substantial amounts of the time and money companies normally spend on developing technology, reverse engineering is a low cost method of modifying an original program to create a competing program.[32]

---

20. *Id.*

21. E.F. Johnson Co. v. Uniden Corp. of Am., 623 F. Supp. 1485, 1488 (D. Minn. 1985).

22. Robert Steinberg, *Microcode—Idea or Expression?* 9 COMPUTER/L.J. 61, 62 (1989). Each instruction, provided to the microprocessor by external software or a human programmer, is converted into about four microinstructions. *Id.*

23. *See* Menell, *supra* note 18, at 1055. *See also* Steven R. Englund, Note, *Idea, Process, or Protected Expression?: Determining the Scope of Copyright Protection of the Structure of Computer Programs*, 88 MICH. L. REV. 866, 870 (1990). Programmers frequently use flowcharts to define the tasks, and then convert the elements of the flowchart into code. Final steps in the process of developing programs include debugging, or testing the program for accuracy and proper function, and documentation of materials that explain the program's functions to the user. Menell, *supra* note 18, at 1051.

24. *See* Aufrichtig, *supra* note 10, at 341.

25. Atari Games Corp. v. Nintendo of Am. Inc., 975 F.2d 832, 844 (Fed. Cir. 1992).

26. Disassemblers are widely available. CONGRESS, *supra* note 14, at 147.

27. *See id.* at 146-50. Decompilation is similar to reverse engineering and is "often characterized as any technique that is used to transform 'machine readable' code into 'human readable' code." *Id.* at 148.

28. McCabe & Tanenbaum, *supra* note 7, at 2.

29. *Id.*

30. "[C]lean room is a metaphor for a software development workplace uncontaminated by familiarity with the expression of a competitor's product." CSTB, *supra* note 3, at 79. Firms use "clean rooms" as a means to avoid charges of infringement, or at least improve the chance of prevailing against such charges. *Id.* at 77.

31. *See* Walter A. Effross, *Legal Risks in Software Protection*, N.J. LAW J., July 12, 1993 at 26, *available in* LEXIS, Nexis Library, NJLAWJ File.

32. CSTB, *supra* note 3, at 78. Reverse engineering is controversial in the industry because it can result in the original program creator having "his lead time erased, his price undercut, and his market reduced for the very thing he created." *Id.* (quoting Howard G. Figueroa, Vice President, Commercial and Industry Relations, IBM Corp.).

B. *Computer Programs and Copyright Law*

1. The Copyright Act of 1976

The U.S. Constitution gives Congress the authority "[t]o promote the Progress of Science and useful Arts, by securing for limited Times to Authors and Inventors the exclusive Right to their respective Writings and Discoveries."[33] Congress enacted the first copyright law in the United States in 1790, with revisions enacted in 1831, 1870, 1909,[34] and most recently, 1976.[35] As of 1974, copyright laws did not specifically cover computer uses. This prompted Congress to form the National Commission on New Technological Uses of Copyrighted Works (CONTU).[36] CONTU's purpose was to study and make recommendations for computer uses. The CONTU Final Report recommended amendments to the Copyright Act of 1976 (Act), which would make computer programs explicitly copyrightable to the extent they embody the author's original creation.[37] In addition, CONTU recommended that a definition of a computer program be added to section 101.[38] In 1980, Congress adopted most of CONTU's recommendations, specifically applying the Copyright Act to computer programs.[39] Computer programs are copyrightable as literary works under section 101 of the Act;[40] however, the Act protects only the expression of an idea or process, rather than the idea or process itself.[41] For protection of ideas and processes, software developers must look to patent law.[42]

---

33. U.S. CONST. art. I, § 8, cl. 8.

34. H.R. REP. No. 1476, 94th Cong., 2d Sess. 47 (1976), *reprinted in* 1976 U.S.C.C.A.N. 5659, 5660.

35. Copyright Act of 1976, 17 U.S.C. § 101-1010 (1988 & Supp. IV 1992).

36. Act of Dec. 31, 1974, Pub. L. No. 93-573, Tit. II, 88 Stat. 1873.

37. NATIONAL COMMISSION ON NEW TECHNOLOGICAL USES OF COPYRIGHTED WORKS, FINAL REPORT 1 (1978) [hereinafter "CONTU REPORT"].

38. *Id.* at 12. CONTU defined a computer program as "a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result." *Id.*

39. Act of Dec. 12, 1980, Pub. L. No. 96-517, § 10, 94 Stat. 3015, 3028 (1980) (codified as amended at 17 U.S.C. §§ 101-117 (1988)).

40. The Act defines literary works as works, "other than audiovisual works, expressed in words, numbers, or other verbal or numerical symbols or indicia, regardless of the nature of the material objects, such as books, periodicals, manuscripts, phonorecords, film, tapes, disks, or cards, in which they are embodied." 17 U.S.C. § 101 (1988).

41. "In no case does copyright protection for an original work of authorship extend to any idea, procedure, process, system, method of operation, concept, principle, or discovery, regardless of the form in which it is described, explained, illustrated, or embodied in such work." 17 U.S.C. § 102(b) (1988); *see also* H.R. REP. No. 1476, 94th Cong., 2d Sess. 47 (1976), *reprinted in* 1976 U.S.C.C.A.N. 5659, 5670 ("Section 102(b) is intended . . . to make clear that . . . the actual processes or methods embodied in the [computer] program are not within the scope of the copyright law.").

42. Although computer programs initially were not considered patentable, Gottschalk v. Benson, 409 U.S. 63 (1972) (interpreted as rendering software unpatentable because algorithms not patentable), patent protection was established by the Supreme Court in 1981. Diamond v. Diehr, 450 U.S. 175 (1981) (use of computer program in rubber-curing process did not render the invention unpatentable). Patent law protects "any new and useful process, machine, manufacture or composition of matter, or any new and useful improvement thereof." 35 U.S.C. § 101 (1988). In addition, the subject matter of the work must be nonobvious. 35 U.S.C. § 103 (1988). Patents provide protection to owners and give the inventor the right to exclude others from making, using, or selling the patented art. Menell, *supra*

2. Interpretation of the Act

Shortly after the passage of the Copyright Act of 1976, litigation involving copyright infringement of computer programs began.[43] One of the first significant cases was *Apple Computer, Inc. v. Franklin Computer Corp.*,[44] in which Apple, one of the computer industry's leaders, sued Franklin, a manufacturer who had designed an "Apple compatible" computer.[45] The U.S. Court of Appeals for the Third Circuit reversed the lower court's denial of a preliminary injunction, and enjoined Franklin from marketing fourteen infringing computer programs.[46] Citing the CONTU Report, the court held that a computer program is copyrightable as a literary work, whether expressed in object code or source code.[47] In addition, the court reaffirmed that a computer program embedded in a memory device, such as a chip, meets the Act's requirement of fixation, and is therefore copyrightable.[48] Finally, the court refused to per se exclude from copyright an operating system program, or a program put to a utilitarian, or functional, use.[49]

3. The Idea/Expression Dichotomy

Although the Act clearly establishes copyright protection for computer programs, the extent of protection is less clear. A major area of discussion in most copyright infringement cases centers on where to draw the line on the "gradual sloping beach called the merger of idea and expression."[50] Section 102(b) of the Act clearly precludes protection of ideas, but many courts have found the line between idea and expression to be ambiguous.[51] Judge Learned Hand, one of the most cited authorities

---

note 18, at 1074. The number of software related patents awarded has risen from none in 1980 to about 200 annually. CSTB, *supra* note 3, at 31.

43. *Eg.*, Synercom Technology, Inc. v. University Computing Co., 462 F. Supp. 1003 (N.D. Tex. 1978) (claim of copyright infringement of instruction manuals and input formats used with computer program); Data Cash Systems, Inc. v. JS&A Group, Inc., 480 F. Supp. 1063 (N.D. Ill. 1979), *aff'd*, 628 F.2d 1038 (7th Cir. 1980) (claim of copyright infringement of object code).

44. 714 F.2d 1240 (3d Cir. 1983), *cert. dismissed*, 464 U.S. 1033 (1984).

45. *Id.* at 1242-43.

46. *Id.* at 1242.

47. *Id.* at 1249; *see also* Johnson Controls, Inc. v. Phoenix Control Sys., Inc., 886 F.2d 1173, 1175 (9th Cir. 1989) (source and object code consistently held protected by copyright); MARSHALL A. LEAFFER, UNDERSTANDING COPYRIGHT LAW § 3.6(B) (1989).

48. *Apple Computer*, 714 F.2d at 1249; *see also* Williams Elecs., Inc. v. Artic Int'l, Inc., 685 F.2d 870, 876 (3d Cir. 1982) (program embedded in ROM device copyrightable). The Copyright Act requires that a work be "fixed in any tangible medium of expression." 17 U.S.C. § 102(a) (1988 & Supp. IV 1992).

49. *Apple Computer*, 714 F.2d at 1252 (citing Apple Computer, Inc. v. Formula Int'l, Inc., 562 F. Supp. 775, 780 (C.D. Cal. 1983), *aff'd*, 725 F.2d 521 (9th Cir. 1984)); *see also* Mazer v. Stein, 347 U.S. 201, 218 (1954) ("We find nothing in the copyright statute to support the argument that the intended use or use in industry of an article eligible for copyright bars or invalidates its registration."); CONTU REPORT, *supra* note 37, at 21 ("That the words of a program are used ultimately in the implementation of a process should in no way affect their copyrightability.").

50. CSTB, *supra* note 3, at 23; *see also, e.g.*, Frybarger v. International Business Mach. Corp., 812 F.2d 525, 529-30 (9th Cir. 1987) (discussing which features of two similar video game programs are protected ideas).

51. *See, e.g.*, *Apple Computer*, 714 F.2d at 1253.

on the subject, noted that "[t]he test for infringement of copyright is of necessity vague."[52] In his opinion in *Nichols v. Universal Pictures*,[53] Judge Hand described an "abstractions" test which separates the idea from the expression by analyzing the patterns of increasing generality in a written or dramatic work.[54] In this "series of abstractions," the work reaches a point where it is no longer protected because it consists of the idea only.[55]

Further difficulty in drawing the line between the idea and its expression arises because many, if not all, computer programs are essentially utilitarian rather than artistic works. In considering this issue, many courts have relied upon the rationale of the *Baker v. Selden* decision.[56] In *Baker*, the Court held that a ledger form in an accounting book was not copyrightable because the form was necessary and incidental to the method of accounting.[57] Therefore, the utilitarian nature of the work rendered it unprotected. Courts have also expressed this idea as the *scenes a faire* doctrine, where expressions that are indispensable in the standard treatment of an idea are not protected.[58] Similarly, the court in *Apple Computer* focused on whether the idea can be expressed in various modes. If the idea can only be expressed in one way, the idea and expression "merge" and copyright protection is unavailable.[59]

Another issue in protectable expression is whether protection extends beyond the literal code in a program. In *Whelan Associates, Inc. v. Jaslow Dental Laboratory, Inc.*,[60] the U.S. Court of Appeals for the Third Circuit held that the structure, sequence, and order of a program is protectable as expression.[61] According to the court, a dental laboratory record keeping program infringed a similar program, which accomplished the same purpose, but was written in a different language.[62] Even though the literal code of the two programs differed, the court found infringement because

---

52. Peter Pan Fabrics, Inc. v. Martin Weiner Corp., 274 F.2d 487, 489 (2d Cir. 1960).

53. 45 F.2d 119 (2d Cir. 1930), *cert. denied*, 282 U.S. 902 (1931).

54. *Id.* at 121.

55. *Id.*; *see also* William E. Hilton, *Quantifying Originality: A Logical Analysis for Determining Substantial Similarity in Computer Software Copyright Infringement Actions*, 31 IDEA 269, 285-86 (1991) (describing the abstractions analysis).

56. 101 U.S. 99 (1879).

57. *Id.* at 104.

58. Copyright protection cannot "be afforded to elements of expression that necessarily follow from an idea." Data E. USA, Inc. v. Epyx, Inc., 862 F.2d 204, 208 (9th Cir. 1988) (no infringement where idea inseparable from expression of home computer karate game). *See also* See v. Durang, 711 F.2d 141, 143 (9th Cir. 1983) (scenes from a play which were "stock scenes" or those which "flowed necessarily from common unprotectable ideas" not protected).

59. *Apple Computer, Inc.*, 714 F.2d at 1252. Another court has stated, "at some point in the process the idea or 'broad and general statement of the purpose' of the program merges into the expression, the 'smaller and more detailed tasks' necessary to carry out that idea". E.F. Johnson Co v. Uniden Corp. of Am., 623 F. Supp. 1485, 1501-02, n.17 (D. Minn. 1985). *See also* CONTU REPORT, *supra* note 37, at 20 ("[W]hen specific instructions, even though previously copyrighted, are the only and essential means of accomplishing a given task, their later use by another will not amount to an infringement."); Englund, *supra* note 23, at 877 n.56 (explaining the merger doctrine).

60. 797 F.2d 1222 (3d Cir. 1986), *cert. denied*, 479 U.S. 1031 (1987).

61. *Id.* at 1240.

62. *Id.*

the structure and organization of the two programs were substantially similar.[63] Commentators and courts, however, have since criticized the *Whelan* decision as being overbroad in the protection it granted computer programs.[64]

The Supreme Court in 1991 significantly narrowed this broad protection in *Feist Publications, Inc. v. Rural Telephone Service Co.*.[65] Although *Feist* concerned the copyrightability of telephone directory white pages, the Supreme Court's holding that copyright law did not protect Rural's listings had an impact on software copyrightability as well.[66] The *Feist* Court rejected the "sweat of the brow" doctrine,[67] refusing to allow copyright protection for factual compilations absent the creation of an original work in the selection and arrangement of pre-existing materials.[68] Although the Court conceded that factual compilations may possess the originality required for copyright protection,[69] the Court limited protection "only to those components of a work that are original to the author."[70] The decision therefore lowered the threshold of protection from *Whelan*, which allowed protection for almost every aspect of a program except the broad idea of running a dental laboratory efficiently.[71] Because computer programs consist of a number of subroutines and ideas, they normally contain more than one broad idea. When an idea is broken into component subroutines, many aspects of the program are more utilitarian in nature, and therefore may not be copyrightable.[72] For these reasons, the *Feist* decision has allowed software developers more freedom in building on the work and ideas of others.[73]

4. Substantial Similarity

Having determined the parts of a program that are protected by copyright, a court must then determine whether the programs are substantially similar. Different circuits have varied somewhat in their formulation of tests for substantial similarity. One test is an "iterative" approach, which

---

63. *Id.* at 1248.

64. *See, e.g.*, Computer Assoc. Int'l, Inc. v. Altai, Inc., 982 F.2d 693, 706 (2d Cir. 1992) (noting *Whelan*'s "outdated appreciation of computer science"); Sega Enters. Ltd. v. Accolade, Inc., 977 F.2d 1510, 1525 (9th Cir. 1992) (criticizing *Whelan* as simplistic and overbroad); CLAPES, *supra* note 12, at 100 (discussing the immediate criticism of the decision); Steven W. Lundberg et al., *Identifying Uncopyrightable Computer Implemented Processes and Systems*, 9 COMPUTER LAW. April 1992, at 7, 8 (discussing the *Whelan* court's "misconception" of processes embodied in computer programs); Menell, *supra* note 18, at 1082-83 (discussing increased costs of innovative activity as a result of broad copyright protection).

65. 111 S. Ct. 1282 (1991).

66. *See generally* Philip D. Bartz & Jonathan Band, *Feist v. Rural Telephone: The Beginning of the End of Software Overprotection?*, 8 COMPUTER LAW. 10 (1991).

67. Some courts had developed the notion "that copyright was a reward for the hard work that went into compiling facts." *Feist*, 111 S. Ct. at 1291.

68. *Id.* at 1293 (citing 17 U.S.C. § 101 (1988)).

69. *Id.* at 1287.

70. *Id.* at 1289.

71. *See* Bartz & Band, *supra* note 66, at 11.

72. *See Sega Enters. v. Accolade Inc.*, 977 F.2d 1510, 1525 (9th Cir. 1992) (emphasizing the essentially utilitarian nature of computer programs).

73. Bartz & Band, *supra* note 66, at 11.

was applied in *E.F. Johnson Co. v. Uniden Corp. of America*.[74] In that case, the court characterized the test for substantial similarity as having two prongs.[75] The court first required proof that the defendant used the copyrighted work in preparing a copy.[76] This was established by proof of access to the work and similarity sufficient to infer use of the copyrighted work.[77] Next, the court required that the alleged copy be an iterative production, or one produced by an exact duplication of substantial parts of the copyrighted work.[78] The court noted that expert testimony is necessary to analyze the similarities in both prongs of the test, in addition to the lay observer's impressions of the "total concept and feel" of the two works.[79]

A more sophisticated test which has been used by several circuits, is an extrinsic/intrinsic test.[80] In *Whelan*, the U.S. Court of Appeals for the Third Circuit discussed this type of two part test for substantial similarity.[81] First, by deciding whether the works are sufficiently similar, the trier of fact must establish that the alleged infringer used the copyrighted work. In applying the "extrinsic" or objective test, expert testimony may be used to assist the fact-finder.[82] Next, the trier of fact applied the "intrinsic" or subjective part of the test to determine whether the copying was illicit, as viewed from the lay observer's perspective, without expert testimony.[83] The intrinsic part of the test has also been characterized as capturing the "concept and feel" of the works, similar to the iterative approach.[84] This test has also been used in the Fourth and Ninth Circuits.[85]

A variation, used in the First and Seventh Circuits, applies only the second prong of the extrinsic/intrinsic test. This is known as the "ordinary observer" test. Here, the inquiry the court makes is whether an ordinary, reasonable person would conclude that the protected part of a work had been copied.[86]

---

74. 623 F. Supp. 1485, 1492-93 (D. Minn 1985) (citing Williams Elecs. v. Artic Int'l, Inc., 685 F.2d 870 (3d Cir. 1982) as following the iterative approach).

75. *Id.* at 1492.

76. *Id.* at 1493.

77. *Id.*

78. *Id.*

79. *Id.*

80. *E.g.*, Ford Motor Co. v. Summit Motor Prods., Inc., 930 F.2d 277, 291 (3d Cir. 1991); Moore v. Columbia Pictures Indus., Inc., 972 F.2d 939, 945 (8th Cir. 1992).

81. The court discusses this bifurcated approach and hints that it is the law in the Third Circuit but does not adopt or utilize the test in *Whelan* itself. Whelan Assocs. v. Jaslow Dental Lab., 797 F.2d 1222, 1232 (3rd Cir. 1986), *cert. denied*, 479 U.S. 1031 (1987) (describing the test as set forth in Arnstein v. Porter, 154 F.2d 464, 468-69 (2d Cir. 1946)).

82. *Id.*

83. *Id.* at 1232. *See also* Sid & Marty Kroft Television Prods., Inc. v. McDonald's Corp., 562 F.2d 1157, 1164-65 (9th Cir. 1977) (discussing the extrinsic and intrinsic tests of substantial similarity).

84. *See* Dawson v. Hinshaw Music Inc., 905 F.2d 731, 733 (4th Cir.), *cert. denied*, 498 U.S. 981 (1990).

85. *See, e.g., id.* at 732-33 (two part intrinsic/extrinsic test); Data E. USA, Inc. v. Epyx, Inc., 862 F.2d 204, 208 (9th Cir. 1988) (two step test for substantial similarity in the Ninth Circuit).

86. *See* Concrete Machinery Co. v. Classic Lawn Ornaments, Inc., 843 F.2d 600, 607 (1st Cir. 1988) (applying second part of test from Arnstein v. Porter, 154 F.2d 464 (2d Cir. 1946));

More recently, in *Computer Associates International, Inc. v. Altai, Inc.*,[87] the U.S. Court of Appeals for the Second Circuit utilized a three part test for substantial similarity, consisting of an abstraction, filtration, and comparison analysis.[88] The first part uses Judge Hand's abstractions analysis.[89] After determining the various levels of abstractions, the trier of fact must filter unprotectable material from the work. Standard techniques in the public domain are excluded[90], as well as portions of the program necessary because of structural[91] or external factors.[92] The court characterized this filtration process as a method of defining the scope of copyright protection.[93] This process ideally leaves a "core of protectable material."[94] The final step involves comparing this core of protectable expression in the copyrighted work to the alleged copy for substantial similarities.[95] This test somewhat limits copyright protection because by protecting only a core of expression, very little of the work may be left protected. The court's rationale for such limitation, however, emphasized a desire to create incentive for future program development and noted the ultimate aim of copyright law in stimulating artistic creativity for the public good.[96] This three part test has also been adopted by the U.S. Court of Appeals for the Tenth Circuit since the decision in *Computer Associates International*.[97] In general, most of the circuit courts of appeal seem to be shifting away from the test set forth in *Whelan*.[98]

5. Fair Use Exception

Even if a court finds substantial similarity under one of the above tests, copyright law permits some degree of copying under statutory exceptions.[99] One such exception is the traditional privilege of other authors for "fair use" of the copyrighted work.[100] In *Harper & Row Publishers v.*

Evans Newton Inc. v. Chicago Systems Software, 793 F.2d 889, 895 (7th Cir.), *cert. denied*, 479 U.S. 949 (1986); Atari, Inc. v. North Amer. Philips Consumer Elecs. Corp., 672 F.2d 607, 614 (7th Cir.), *cert. denied*, 459 U.S. 880 (1982).

87. 982 F.2d 693 (2d Cir. 1992).

88. *Id.* at 706-11.

89. *Id.* at 706-07. *See also supra* notes 54-55 and accompanying text.

90. *Computer Assocs. Int'l*, 892 F.2d at 710.

91. *Id.* at 707-09.

92. *Id.* at 707-710.

93. *Id.* at 707.

94. *Id.* (citing 3 NIMMER, *supra* note 6, § 13.03(F)(5), at 13-72).

95. *Id.* at 710-11.

96. *Id.* at 711 (quoting Twentieth Century Music Corp. v Aiken, 422 U.S. 151, 156 (1975) (Stewart, J.)).

97. *See* Autoskill Inc. v. National Educ. Support Sys., Inc., 994 F.2d 1476, 1490 (10th Cir.), *cert denied*, 114 S.Ct. 307 (1993).

98. *See* Lee T. Gesmer, *Decisions May Signify a Judicial Turnabout*, NAT'L. L.J., Jan. 19, 1993, at S2, S3 (indicating that *Whelan* is no longer followed in the Second, Fifth, Ninth and Federal Circuits and predicting other circuits will follow *Computer Associates International*.)

99. 17 U.S.C. §§ 107-120 (1988 & Supp. IV 1992) (limitations on exclusive rights).

100. 17 U.S.C. § 107 (1988). "[T]he fair use of a copyrighted work, including such use by reproduction in copies or phonorecords or by any other means specified by that section, for purposes such as criticism, comment, news reporting, teaching (including multiple copies for classroom use), scholarship, or research, is not an infringement of copyright." 17 U.S.C. § 107 (1988). Factors which are considered in determining whether a use is fair include:

Nation Enterprises,[101] the Supreme Court held that a magazine's unauthorized publishing of verbatim quotes from President Ford's memoirs was not a fair use because it deprived the copyright owners of their valuable right to first publication.[102] The Court stated that when a publication is used commercially for profit rather than for a nonprofit educational use, a court will less likely find a fair use.[103] The issue of whether a use is fair, however, requires a case-by-case analysis.[104] In *Sega Enterprises v. Accolade, Inc.*,[105] the court did not allow reverse engineering of a computer program, stating that such a use "for financial gain and . . . aimed at the creation of a competitive product" was not fair use of copyrighted software.[106] This was an important issue which the U.S. Court of Appeals for the Federal Circuit would consider in *Atari* later the same year.

II. INSTANT CASE

A. *Factual History*

Nintendo sells the Nintendo Entertainment System (NES), a home video game system. The system includes a monitor, a console, and controls. To play a game, the user inserts a game cartridge into the console, thereby creating images on the display.[107] In order to prevent cartridges other than Nintendo cartridges from working in the NES, Nintendo designed a security system called the 10NES. The console and authorized game cartridges both contain microprograms,[108] which are chips programmed with the 10NES. The console contains a master chip which acts as a lock to the NES. Authorized game cartridges contain a "slave chip", which acts as a key to the system. When one inserts an authorized cartridge into the NES, the slave chip "unlocks" the console, allowing the

---

1) the purpose and character of the use, including whether such use is of a commercial nature or is for nonprofit educational purposes; 2) the nature of the copyrighted work; 3) the amount and substantiality of the portion used in relation to the copyrighted work as a whole; and 4) the effect of the use upon the potential market for or value of the copyrighted work.

Id.

101. 471 U.S. 539 (1985).

102. *Id.* at 569.

103. *Id.* at 562; *see also* 3 NIMMER, *supra* note 6, § 13.05(A)(1)(c), at 13-161 to 13-165.

104. H.R. Rep. No. 1476, 94th Cong., 2d Sess. 56 (1976), *reprinted in* 1976 U.S.C.C.A.N. 5659, 5680.

105. 785 F. Supp. 1392 (N.D. Cal.), *aff'd in part, rev'd in part by* Sega Enters. v. Accolade, Inc., 977 F.2d 1510 (9th Cir. 1992).

106. *Id.* at 1398. The district court cited legislative history indicating intermediate copying was not intended by Congress to be a fair use. *Id.* A month after the U.S. Court of Appeals for the Federal Circuit decided *Atari Games Corp. v. Nintendo of Am.*, 975 F.2d 832 (Fed. Cir. 1992), however, the U.S. Court of Appeals for the Ninth Circuit reviewed the district court's decision on appeal. In that case, *Sega Enters. v. Accolade, Inc.*, 977 F.2d 1510 (9th Cir. 1992), the court allowed reverse engineering as a fair use when "disassembly is the only way to gain access to the ideas and functional elements embodied in a copyrighted computer program and where there is a legitimate reason for seeking such access." *Id.* at 1527.

107. Atari Games Corp. v. Nintendo of Am., 975 F.2d 832, 835 (Fed. Cir. 1992).

108. Microprograms are copyrightable as long as the instructions are not completely functional. Steinberg, *supra* note 22, at 62.

user to play the game. An unauthorized game cartridge which sends no unlocking message to the console will therefore not work in the NES.[109]

Beginning in 1986, Atari, one of Nintendo's competitors in the home video game market, attempted to replicate the 10NES system.[110] Atari engineers unsuccessfully attempted to understand the 10NES program by monitoring communication between the chips.[111] They also analyzed the chips by chemically peeling layers from NES chips and microscopically examining the object code.[112] Having failed in these efforts, Atari became a Nintendo licensee in 1987.[113] Atari paid Nintendo to insert Atari's games into 10NES cartridges, allowing Atari to market its games to Nintendo console owners.[114]

Apparently dissatisfied with this arrangement, Atari attempted to decipher the 10NES code again. In early 1988, Atari's attorney lied to the Copyright Office and wrongfully obtained a copy of the 10NES program.[115] The attorney stated in his application to the Copyright Office that Atari was a defendant in an infringement action and needed a copy for that litigation.[116] Upon obtaining the source code from the Copyright Office, Atari engineers were able to correct errors in the transcription of the object code from their microscopic analysis of peeled chips.[117] In this way, the wrongfully obtained copy from the Copyright Office permitted Atari to replicate the 10NES object code.[118]

Atari then developed its own program called the Rabbit program, which performed the same function as the 10NES. Although the Rabbit used a different microprocessor and was programmed in a different language, the two programs generated functionally indistinguishable signals. The Rabbit program therefore gave Atari access to Nintendo console owners, in breach of its licensing agreement.[119]

In December 1988, Atari sued Nintendo for unfair competition, Sherman Act violations, and patent infringement.[120] Then, in November 1989, Nintendo sued Atari for unfair competition, patent infringement, copyright infringement, and trade secret violations. The United States District Court for the Northern District of California consolidated the two cases, preliminarily enjoining Atari from exploiting Nintendo's program.[121]

---

109. *Atari*, 975 F.2d at 836.
110. Atari Games Corp. v. Nintendo of Am. Inc., 1991 WL 57304 at *1 (N.D. Cal. 1991).
111. *Id.*
112. *Id.*
113. *Id.*
114. *Atari*, 975 F.2d at 836.
115. *Id.*
116. The regulations of the Copyright Office permit reproduction of a copy if the Office receives a written request from an attorney, on behalf of either the plaintiff or the defendant, in connection with actual or prospective litigation involving the copyrighted work. 37 C.F.R. § 201.2(d)(2) (1992).
117. *Atari*, 975 F.2d at 836.
118. *Id.*
119. *Id.* at 836-37.
120. *Id.* at 837.
121. *Id.* at 835.

B. *Opinion of the Court*

The United States Court of Appeals for the Federal Circuit sustained the preliminary injunction, enjoining Atari from marketing cartridges containing the Rabbit program. The court concluded that Nintendo demonstrated the 10NES program contained at least some protected expression,[122] and that Atari made unauthorized verbatim copies from the Copyright Office.[123] In addition, the court ruled that Nintendo showed a likelihood of success in proving that Atari infringed the 10NES copyright by copying the source code from the Copyright Office, and that the Rabbit program was substantially similar to the 10NES.[124] Finally, the court found that Nintendo was likely to overcome Atari's assertion of the defense of copyright misuse.[125]

The court based its analysis on traditional requirements for infringement cases, applying law from the Ninth Circuit. Under this law, Nintendo had to show 1) ownership of the 10NES program, and 2) copying of protected expression, either by verbatim copying, or by Atari having access to the copyrighted work, plus substantial similarity between the works.[126]

The first step required the court to identify the protected expression in the 10NES program. The court did this by using Judge Hand's abstractions test[127] and the filtration component of *Computer Associates International*.[128] The court held that the 10NES program contained protectable expression, even though it involved the process of unlocking the system.[129] Nintendo's use of arbitrary instructions arranged in a unique sequence to create an arbitrary data stream was found to be protected after the filtration process. External factors did not dictate the design, and the expression was not taken from the public domain.[130] In addition, the expression of the data stream did not merge with the idea because alternate expressions were available, as shown by expert testimony of different ways to generate it.[131]

Next, the court examined whether Atari had infringed by making verbatim copies from the source code from the Copyright Office.[132] The court concluded that the copies from the Copyright Office, which Atari

122. *Id.* at 840.
123. *Id.* at 841-42.
124. *Id.* at 847.
125. *Id.* Copyright misuse is a defense which "bars a culpable plaintiff from prevailing on an action for infringement of the misused copyright." Lasercomb Am., Inc. v. Reynolds, 911 F.2d 970, 972 (4th Cir. 1990) (explaining copyright misuse defense as inherent in copyright law).
126. *Atari*, 975 F.2d at 837-38.
127. *See supra* notes 53-55 and accompanying text.
128. *See supra* notes 84-94 and accompanying text.
129. Atari Games Corp. v. Nintendo of Am., 975 F.2d 832, 840 (Fed. Cir. 1992).
130. *Id.* at 840.
131. *Id.* In a later opinion, Atari Games Corp. v. Nintendo of Am., Inc., 1993 WL 214886 (N.D. Cal. 1993), the court emphasized that the data stream itself was not protected, only the unique expression. *Id.* at *9.
132. *Id.* at 840.

obtained in violation of its regulations,[133] were unauthorized copies.[134] Since reproducing an unauthorized copy from the Copyright Office violates the Copyright Act, the court found that Nintendo was likely to show infringement.[135]

In dicta, the court said that reverse engineering by Atari to understand the unprotected ideas in the 10NES program was a fair use.[136] The court proceeded to say, however, that fair use does not extend to intermediate copying for commercial exploitation of protected expression.[137] Furthermore, the court stated that in order to invoke the fair use exception, the user of the copyrighted work must have an authorized copy.[138] Since Atari had an unauthorized copy of the program and used reverse engineering beyond that necessary to understand the program, the court refused to find Atari within the fair use exception.[139] In doing so, the court emphasized the importance of copyright in balancing authors' interests in controlling and exploiting their works, and society's interest in the free flow of information and ideas.[140] According to the court, patent law is the appropriate medium to protect one's ideas, and such protection cannot be acquired by writing a program "in an unintelligible format and asserting copyright infringement against those who try to understand" the idea or process.[141]

The court used the two step, extrinsic and intrinsic analysis for substantial similarity.[142] Stating that expert testimony is appropriate to discern similarities in the two programs, the court noted several unnecessary instructions in the Rabbit program. This indicated strong evidence of copying, rather than independent creation.[143] The court therefore found the two programs substantially similar.[144]

Finally, the court rejected Atari's copyright misuse defense. The court drew a parallel to the patent misuse defense and acknowledged that the copyright misuse defense would be appropriate under certain circumstances.[145] Atari, however, was not allowed to raise the equitable defense of copyright misuse because it had "unclean hands" as a result of its attorney lying to the Copyright Office.[146]

---

133. *See supra* note 110 and accompanying text.

134. *Atari*, 975 F.2d at 841.

135. *Id.* at 842.

136. *Id.* at 843-44.

137. *Id.* at 843-44.

138. *Id.* at 843.

139. *Id.* at 844.

140. *Id.* at 842 (citing Sony Corp. of Am. v. Universal City Studios, Inc., 464 U.S. 417, 429-30 (1984)).

141. *Atari*, 975 F.2d at 842.

142. *Id.* at 844.

143. *Id.* at 845.

144. *Id.*

145. *Id.* at 846.

146. *Id.*

III. ANALYSIS

A. *Reverse Engineering*

One of the most significant aspects in the *Atari* opinion is the dicta regarding reverse engineering. The statement that reverse engineering is a fair use to the extent it is necessary to understand a program[147] could have a significant effect on the computer industry. Although courts have previously refused to allow reverse engineering as a fair use,[148] the court in *Atari* expressly allowed reverse engineering to discern the unprotected ideas of a work.[149] The court restricted this use somewhat, requiring the use be of an authorized copy and that it not be used to profit through commercial exploitation.[150] Although allowing reverse engineering narrows the scope of copyright protection for computer software, the court's rationale is logical, and the decision corresponds with the purposes of copyrights as articulated in the U.S. Constitution.

The purpose of copyright law, as stated in the U.S. Constitution, is to promote science and encourage innovation and creativity.[151] Ideally, the extent of protection for computer software involves a balance of public and private interests.[152] Since computer programs are often highly utilitarian works, they may require less protection to promote the progress of technological advances in the public interest. Because technological development involves sequential improvements on existing knowledge, scientists and entrepreneurs often fear that business and legal constraints will slow the progress that comes from shared learning.[153] The computer industry, however, is a highly profitable business,[154] and economic interests of software developers are important as well.

For software developers, the cost of creating programs is high compared to the cost of producing copies of existing programs.[155] Develop-

147. *Id.* at 843.

148. *Id.* at 844; *see also* Sega Enters. Ltd. v. Accolade, Inc., 785 F. Supp. 1392, 1398 (N.D. Cal. 1992). In *Sega*, the court argued against allowing reverse engineering as a fair use, since "[a] competitor who reverse engineers a copyrighted computer program . . . [does so with the sole purpose of inducing] the public to purchase his work rather than the original thereby eliminating the market for the original." *Id.* Chemical peeling of microchips or programming in a clean room, however, was said to be acceptable as a fair use. *Id.* at 1399.

149. "Atari did not violate Nintendo's copyright by deprocessing computer chips in Atari's rightful possession. Atari could lawfully deprocess Nintendo's 10NES chips to learn their unprotected ideas and processes." *Atari*, 975 F.2d at 844.

150. *Id.* at 843.

151. U.S. CONST., art. I, § 8, cl. 8. *See also* CSTB, *supra* note 3, at 17. Copyright law has two goals: 1) to encourage disclosure of intellectual ideas to the public, and 2) to protect intellectual property capable of reproduction. Note, *Copyright Protection of Computer Program Object Code*, 96 HARV. L. REV. 1723, 1739 (1983).

152. Balancing of public and private interests involves a need to protect innovation and a need to share ideas in order to encourage compatibility of computer systems. CSTB, *supra* note 3, at 17. Protection which is too liberal results in monopolies and inhibits creativity. Protection which is too conservative also discourages innovation because creative efforts are easily copied. Menell, *supra* note 18, at 1047-48.

153. CSTB, *supra* note 3, at ix.

154. *See supra* notes 2-3 and accompanying text.

155. Both magnetic and hardcopy (on paper) forms of programs can be easily duplicated. CLAPES, *supra* note 12, at 23. Like most forms of intellectual property, developing

ers, therefore, need some degree of protection to avoid a "free rider" problem,[156] where competitors use reverse engineering or copying to avoid development costs. Nevertheless, the benefits of protection must outweigh the costs. Over-broad protection may cause a risk of monopolies, transaction costs from licensing, copyright and patent agreements, and costs of litigating infringement claims.[157] In addition, society pays a price when protection is too broad, since progress may be inhibited as a result of developers having to waste time and resources on research and development which has already been done by other companies.[158] Software developers, however, derive benefits from copyright protection, since it allows them to freely market their innovative ideas without fear of others profiting from their efforts. In this way, copyright protection encourages innovation. Determining what is too much or too little protection is therefore difficult, but the bottom line is that the costs to society clearly outweigh the economic benefits companies receive when protection becomes too broad.

Allowing software programmers to reverse engineer another's work in order to understand the technology of the ideas and processes does not allow them a "free ride." Rather, a company must spend a significant amount of time and resources to decompile a program, as Atari's extensive reverse engineering efforts and difficulty in deciphering Nintendo's code illustrates.[159] Although allowing reverse engineering as a fair use may encourage development of more "security systems" like the 10NES as developers attempt to protect their programs, this is not necessarily a negative effect. On the contrary, it may foster new and creative methods of programming and decompiling code. In addition, reverse engineering encourages innovation by allowing programmers to understand previous work and build upon it. As the court in *Atari* states, "[w]here the infringement is small in relation to the new work created, the fair user is profiting largely from his own creative efforts rather than free-riding on another's work."[160]

An important limitation on reverse engineering is that it not be used "to exploit commercially or otherwise misappropriate protected expres-

---

computer programs involves most expenditure of resources in the development stage (design and implementation). "Reproducing it is trivial. Building it . . . is the hard part." CSTB, *supra* note 3, at 44 (quoting Randall Davis). *See also* Aufrichtig, *supra* note 10, at 342 (cost of developing programs is great); Note, *Copyright Protection of Computer Program Object Code*, 96 HARV. L. REV. 1723 (1983) (copying is easy and inexpensive).

156. Closely related to the free rider problem is the "public goods" problem, which is inherent in markets for intellectual property. The problem stems from the difficulty in excluding anyone from using the work who did not pay for it, and from nonrivalrous competition—additional consumers do not deplete the quantity available to others. A means to correct this problem is to provide limited protection to works containing original expression. This increases the reward to those engaged in intellectual work, and therefore encourages inventive activity. Menell, *supra* note 18, at 1059.

157. *Id.* at 1065.

158. *See id.* at 1083.

159. Atari Games Corp. v Nintendo of Am. Inc., 1991 WL 57304, at *1 (N.D. Cal. 1991).

160. Atari Games Corp. v. Nintendo of Am., 975 F.2d 832, 843 (Fed. Cir. 1992) (quoting New Kids on the Block v. News Am. Pub. Inc., 971 F.2d 302, 307 n. 6 (9th Cir. 1992)).

sion."[161] A problem with the rationale of this argument is that any company using reverse engineering to understand a work is probably involved in ultimately creating their own product for financial gain. The court in *Sega* has addressed this issue since *Atari*; however, it stated that an analysis considering only production of a competing product is too simple.[162] Rather, the court "must consider other aspects of 'the purpose and character of the use' as well."[163] *Sega* indicates that other factors which increase the likelihood of a court finding a fair use include: a direct purpose of simply studying the functional requirements of a program; a public benefit derived from the use, such as increased compatibility[164] or increased amount of creative programs available to the public; and a purpose of competing in the market by developing programs compatible with other systems.[165] These are significant qualifications which help justify allowing reverse engineering.

B. *Protectable Expression and Substantial Similarity*

In ascertaining protected expression and substantial similarity, the court in *Atari* logically applied traditional standards. Although these standards continue to be ambiguous, the court provided some helpful guidelines. The court adopted the tests used in the Ninth Circuit for both issues, along with some aspects of the test used in the Second Circuit for substantial similarity.[166] Among the factors the court used to separate the idea from the expression are similarities beyond that necessary for the function of the program,[167] or common errors in the program that indicate copying.[168] Furthermore, in discerning protected expression, the court also combined aspects of Judge Hand's abstractions analysis and the filtration analysis used in the Second Circuit to provide a method for distinguishing protected aspects of a program.[169] For instance, the court granted protection to a unique and creative sequence of code instructions which Nintendo created independently.[170] In addition, because no external factors dictated code instructions to make the sequence necessary to the function of the program, the court granted copyright protection.[171] The court also considered programs not in the public domain as copyrightable,[172] as well as a particular expression of an idea, where alternate

161. *Id.* at 844.
162. Sega Enters. Ltd. v. Accolade, Inc., 977 F.2d 1510, 1522 (9th Cir. 1992).
163. *Id.*
164. Compatibility in computer systems is considered by many to be an important goal in the software industry. Others assert, however, that the argument for compatibility is a "guise for abetting widespread copying of successful products." CSTB, *supra* note 3, at 73.
165. *Sega*, 977 F.2d at 1522-23. In addition, *Sega* emphasizes that where several ways of expressing an idea exist, the creative expression will be protected, as in *Atari*. Where the expression is purely functional of the idea, however, it will not be protected. *Id.* at 1524 n.7.
166. *Atari*, 975 F.2d at 845.
167. *Id.*
168. *Id.* (citing M. Kramer Mfg. Co. v. Andrews, 783 F.2d 421, 446 (4th Cir. 1986)).
169. *Atari*, 975 F.2d at 839.
170. *Id.* at 840. Nintendo's unique expression of a data stream was protected. *Id.*
171. *Id.*
172. *Id.*

expressions of the idea or process are available.[173] These ideas are traditional in the development of copyright law.[174] In general, the *Atari* opinion indicates a shift in the analysis of copyright protection from that of *Whelan*[175] to that of *Computer Associates International*.[176] This analysis provides less protection, but it may be the most appropriate for computer programs.

C. *Implications for the Software Business*

Due to this narrowing of protection for computer programs, one consequence of *Atari* may be that software developers will continue to attempt to protect their products with methods other than copyright. Other possible methods are patent, trade secret, misappropriation, and trademark law.[177] Although CONTU thought copyright the best medium,[178] considering the utilitarian characteristics of most computer programs, patent law may be more appropriate today.[179] Due to rapid technological advances in the industry, however, patent protection is not viable for most programs.[180] Patent law requires a novel, non-obvious innovation, compared to the less stringent originality requirement of copyright law, and the process for obtaining a patent is much more costly and time consuming than for a copyright.[181] Obtaining a patent also requires disclosure of the invention to the patent office, which may destroy availability of trade secret protection.[182]

Although patent law is therefore not the solution for most businesses, several other options exist to protect software, including trade secret, misappropriation, and trademark law. Trade secret law requires that the information be novel and valuable in the trade or business, and that it be secret.[183] Most firms utilize this type of protection with license agreements.[184] One drawback with this option is that a licensee can reverse engineer and reassemble software to a form bearing little resemblance to the original, which defeats trade secret protection.[185] Another option is misappropriation, a branch of unfair competition.[186] Finally, trademark pro-

---

173. *Id.*
174. *See supra* notes 50-73 and accompanying text.
175. *See supra* notes 60-64 and accompanying text.
176. *See supra* notes 84-95 and accompanying text.
177. Menell, *supra* note 18, at 1077.
178. *See* CONTU Report, *supra* note 37, at 17.
179. Copyright law is necessarily limited for computer programs because they are utilitarian works. *See* Englund, *supra* note 23, at 893.
180. Over 90% of computer programs probably fail to meet the non-obvious requirement, and are therefore not patentable. 1 DAVID BENDER, COMPUTER LAW § 3A.02[1] (1993).
181. McCabe & Tanenbaum, *supra* note 7, at 4. Patents take several years to obtain, and most computer programs become obsolete before protection is available. Note, *supra* note 152, at 1743 and notes 149-153. Owners of copyrights are protected for the life of the author plus 50 years, or 75 years from publication for works created for hire. 17 U.S.C. § 302(a), (c) (1978). In contrast, patents only protect a work for 17 years. 35 U.S.C. § 154 (1988).
182. Menell, *supra* note 18, at 1076.
183. *Id.* at 1077.
184. CSTB, *supra* note 3, at 29.
185. *Id.* at 30.
186. Misappropriation protects collected information. Menell, *supra* note 18, at 1078.

tection may help the developer by protecting the reputation one gains from introducing a product first.[187] Such reputation is potentially valuable, depending on how quickly competitors can reverse engineer and create a similar program.

Although all the options for protection have both advantages and disadvantages, *Atari* will probably force software companies to move away from copyright as a sole means of protection. Due to the lack of availability of patents for most software programs and the minimal protection provided by other options, however, allowing reverse engineering as a fair use creates a problem for software developers who wish to protect their work. The court reasonably applied copyright law in *Atari*, but the decision fails to solve the problem of protecting computer software in a way which balances all interests. Neither copyright law nor patent law ideally suits the unique characteristics of computer programs as artistic works and utilitarian processes.[188] Ultimately, software developers simply need a clearer picture of the law to provide a better solution to protect their valuable creative efforts.

CONCLUSION

After *Atari*, software companies do not necessarily have a clear picture of copyright law as it applies to their programs. The U.S. Court of Appeals for the Federal Circuit continued the tradition of applying vague standards in determining protected expression for computer programs. This area of the law, however, may necessarily remain ambiguous due to the unique nature of computer software and intellectual property in general.[189] More significantly, the *Atari* decision narrowed copyright protection available for computer programs[190] by stating in dicta that reverse engineering is a fair use to understand the ideas and processes of a program.[191] In this way, the court has allowed software developers to cut costs significantly in developing programs to compete with existing programs.[192] In attempting to balance the interests of society and those of entrepreneurs and software developers, the court has given developers little option but to seek other areas of the law to protect their work. Patent law is available to protect useful, novel, and non-obvious ideas, but the time and cost of obtaining patent protection make it an inviable option for most software developers.[193] Thus, while the *Atari* decision may have promoted the purpose of copyright in allowing the free flow of ideas, allowing

187. Trademark law protects marks used by manufacturers to aid consumers in avoiding confusion. *Id.* at 1078, 1081.

188. "Old doctrines and principles are being stretched to unprecedented dimensions to accommodate developing software copyright issues." David Goldberg & Robert J. Bernstein, *Confrontation With the Computer Age*, N.Y. LAW J., Sept. 17, 1993 at 3, 29.

189. *See supra* notes 50-59 and accompanying text.

190. *See* Ellis, *supra* note 7, at 79.

191. The idea that reverse engineering, or disassembly, is a fair use as a matter of law was expressly upheld in Sega Enters. Ltd. v. Accolade, Inc., 977 F.2d 1510, 1514 (9th Cir. 1992).

192. *See* CSTB, *supra* note 3, at 78.

193. *See supra* notes 180-83 and accompanying text.

reverse engineering may have created a problem for computer program developers in protecting their valuable innovations.

Susan E. Dallas